

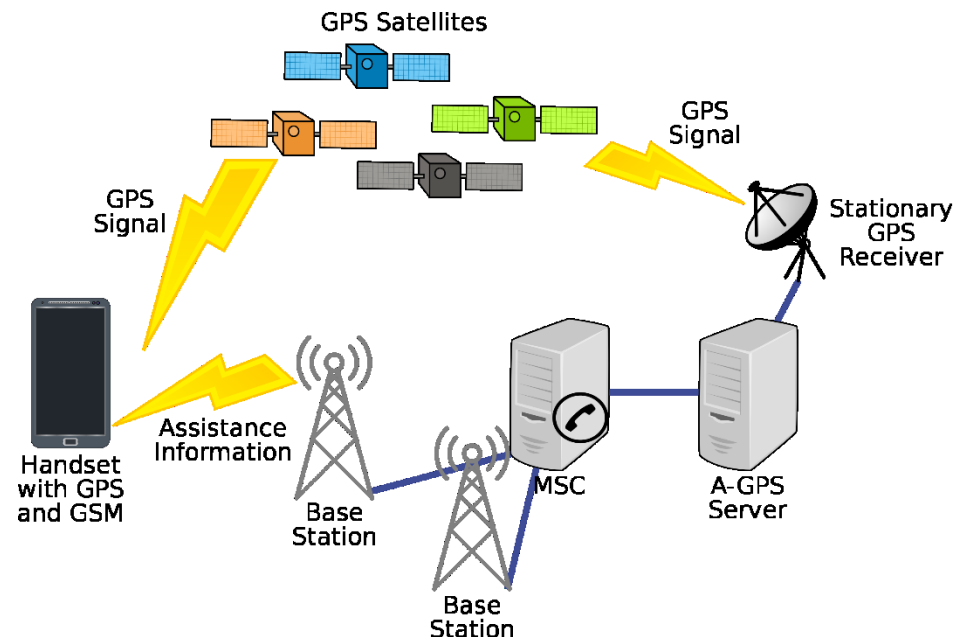
Wireless Networks for Mobile Applications

Prof. Claudio Palazzi
cpalazzi@math.unipd.it

Indoor Localization

Localization

- Quite easy outdoor
 - GPS and A-GPS (Assisted GPS)
- Much harder indoor



Location-Aware Applications

- Subset of context-aware applications
- Wide range of opportunities due to the popularity of smartphones
- Even indoor applications
 - Need for indoor localization (still a challenge)

Metrics

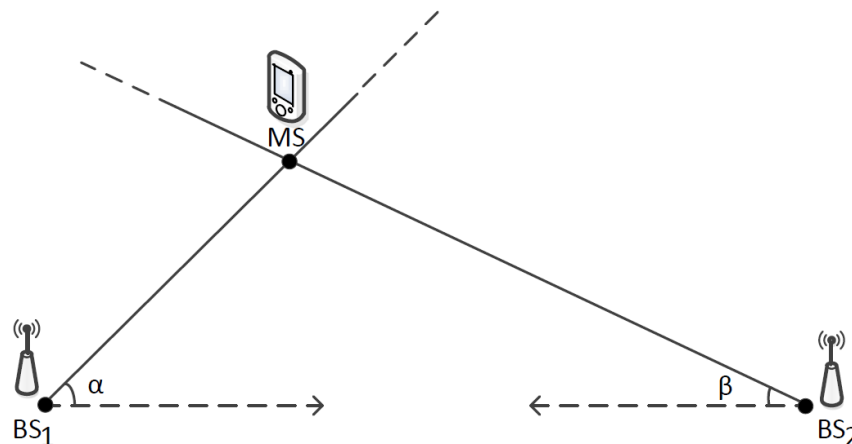
- To assess a technology for indoor localization we can use several metrics
 - **Accuracy:** average error between estimated measure and the actual one
 - **Precision:** error distribution regarding the actual position vs the estimated one
 - **Robustness:** the ability in maintaining accurate estimation even when changing the context/environment
 - **Scalability:** the system behavior when changing the number and density of devices
 - **Cost:** includes the hardware, the initial set up, the maintenance...

Notation

- We can assume to use a cartesian coordinate system in the monitored environment
- Mobile Station (MS): is the device that needs to be localized
 - E.g., a smartphone
- Base Station (BS): is an infrastructure component of the coordinate system
 - E.g., an Access Point

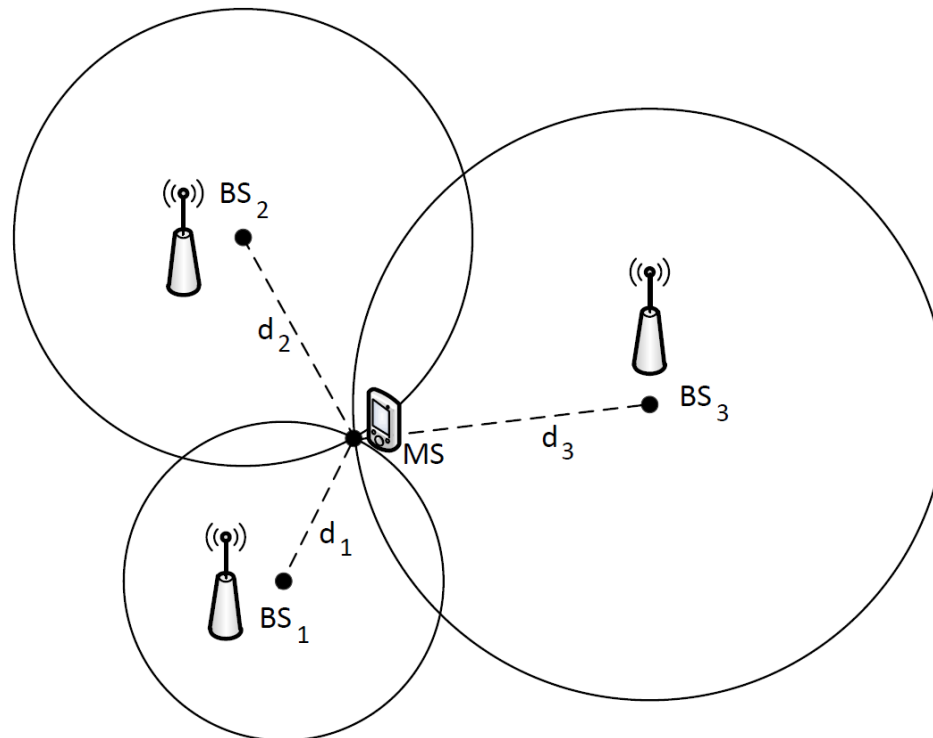
Triangulation

- The Triangulation method (or «Angle of Arrival») requires knowledge of the arrival angles of the signal emitted by the MS and received by the BS
 - At least two angles are needed to compute MS position
 - Requires complex hardware on the BS
 - Not really usable indoor since strong multipath effects in indoor environments



Trilateration

- Trialateration requires knowledge of the distance between the MS and the BS
 - Required the distance with 3 or 4 BS's for localization in 2D or in 3D, respectively

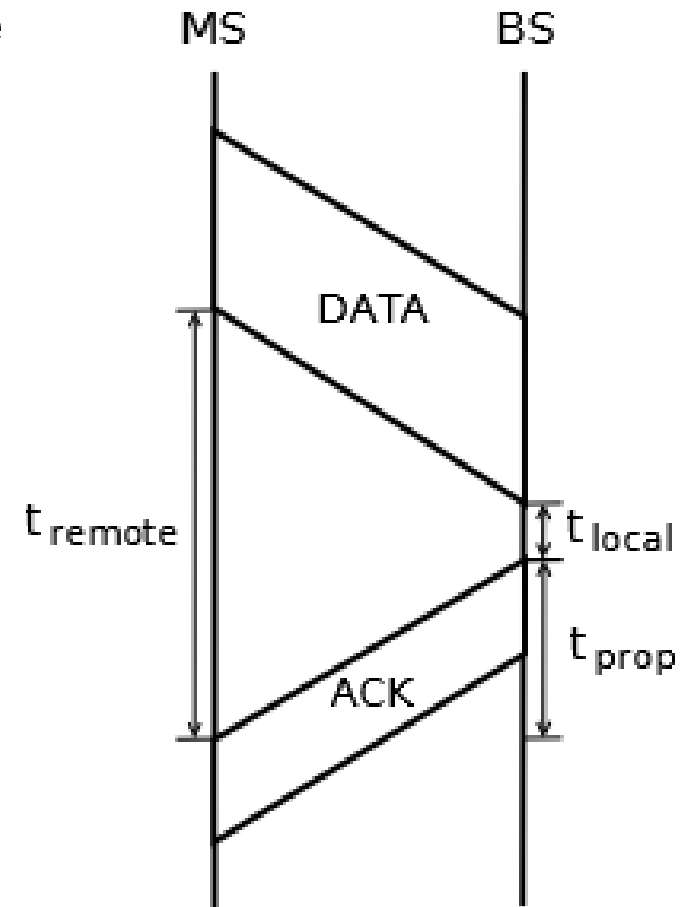


Distance Estimation

- The distance between MS and BS can be estimated through the propagation time of the radio signal
- Since they are electromagnetic waves the propagation speed is $c = 3 \times 10^8 \text{m/s}$.
 - Considering t_{prop} the measured propagation time, then the distance is $d = c \times t_{\text{prop}}$
- Time Of Arrival (TOA)
 - Requires MS and BS to have a synchronized clock and the possibility to exchange data
 - 1) BS emits the signal and sends to the MS the time t_1 at which the transmission ended
 - 2) The MS completes the reception of the signal at time t_2
 - 3) The MS computes the propagation time as $t_{\text{prop}} = t_2 - t_1$

RTT – Round Trip Time

- Does not require data exchange or clock synchronization
 - Measures the time required for the path MS-BS-MS
 - $t_{\text{prop}} = (t_{\text{remote}} - t_{\text{local}}) / 2$
 - t_{local} is variable as it depends on the reaction time of the hardware
 - This error cannot be avoided

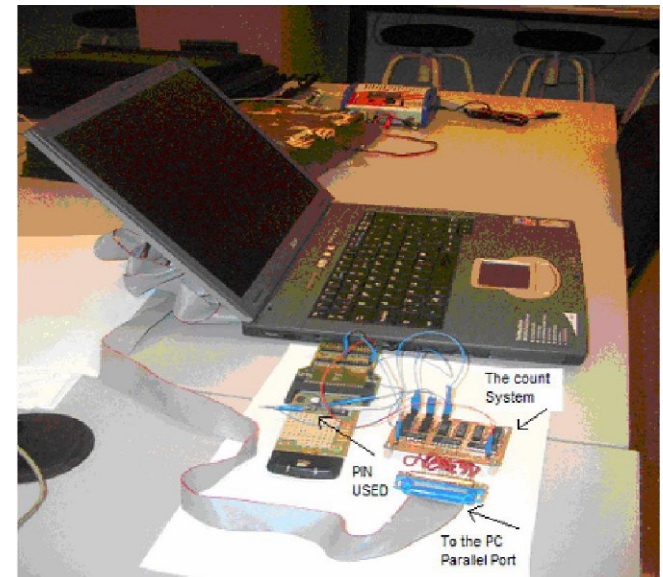


Measurement Error

- We need to consider the measurement error generated by the granularity of the timer used to measure the time
- Most WLAN 802.11 boards allow to save the hardware timestamp of MAC layer packets with a precision of $1\mu\text{s}$
 - Corresponding to a granularity of 300 m when considering the speed of light
- This precision is not sufficient and two possible solutions are proposed:
 - **HW approach**: use the time stamp provided by enhanced/modified HW
 - **SW approach**: use multiple measurements to obtain an estimation close to the actual value

Hardware Approach

- Specific hardware able to use bits transmitted/received to trigger a MAC layer counter based on the clock of the WLAN board
 - This clock has a frequency of 44 MHz, which corresponds to a precision of $2.27 \times 10^{-8} \text{ s} \rightarrow 6.82 \text{ ns}$
 - Improved precision via statistical methods and multiple measurements
- Measurements are done at the lowest possible layer (MAC layer) to avoid variable delays in execution time induced by upper layers
- With 802.11 the RTT is measured with the Data-Ack pair of packets as the time interval between the reception of the former and the transmission of the latter is reasonably constant (SIFS)

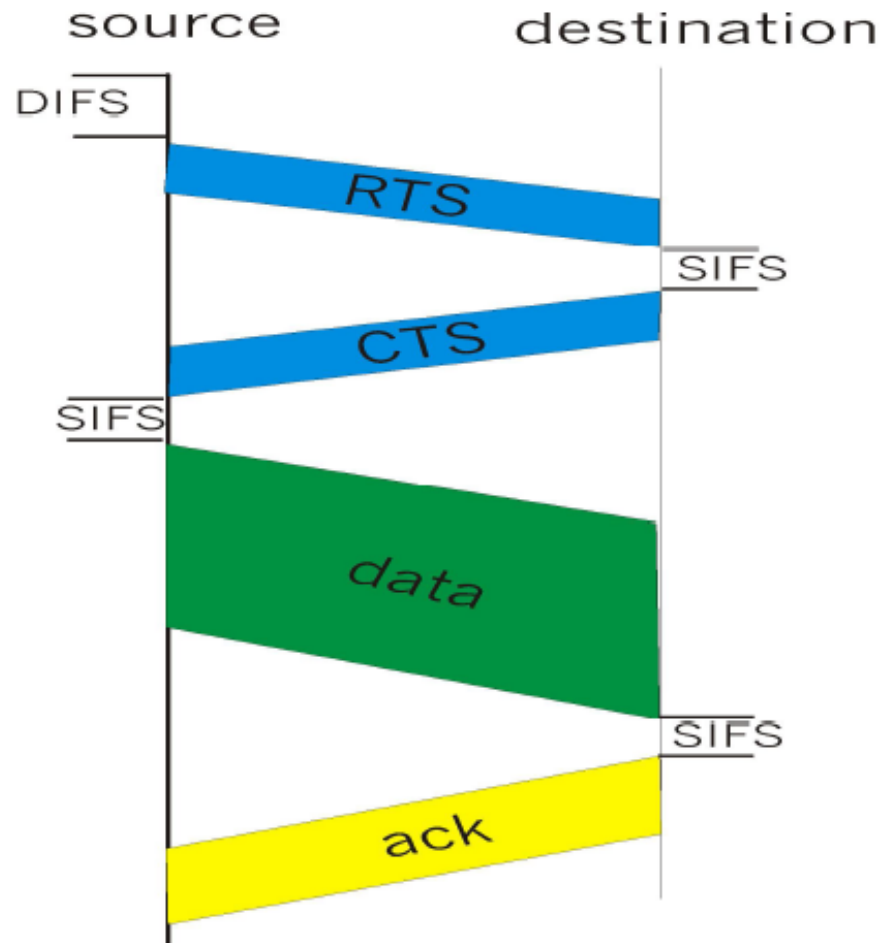


Software Approach

- Hardware time-stamps are provided by regular WLAN boards only for received packets (not for sent ones)
- In case of a pure software approach we hence need to introduce a monitoring station as close as possible to the MS that we want to localize
- This monitoring station has to listen to communications between MS and BS to obtain consistent hardware time-stamps both for sent and received packets (by the MS)
- Requiring a monitoring station makes this approach unfeasible for practical purposes beside the academic field

Software Approach: Goodtry

- **Goodtry** is a prototype developed for research purposes. It measures sequences of packets RTS-CTS-Data-Ack, hence measuring twice the RTT for each transmission. The positioning is done through the lowest weighted squared errors which already includes a method to manage multiple measurements



TDOA – Time Difference of Arrival

TDOA measures the arrival time of a signal emitted by the MS towards multiple BS's and exploits the difference among the arrival times to extract the position of the MS.

- Requires synchronization among BS's
- For 3D localization we need at least 4 BSs's, whereas for 2D we need 3
- Requires a location server to manage both the synchronization and the measurement collection from the various BS's

Not usable for self-positioning: measurements are possible only at the BS

Scene Analysis

Scene analysis is a method composed by two main phases:

- 1) Collection (and storage in a database) of **fingerprints** of the scene in predetermined and known locations; generally a fingerprint includes the measurement of the signal strength received by a MS (RSSI) related to different BS's
- 2) Collection of the fingerprint related to the current position (not known) and comparison with data in the database through artificial intelligence algorithms (e.g., k-NN, SVM, . . .) or statistical methods

Even with this method, we need at least 4 BS's for 3D localization and at least 3 BS's for 2D

RSS – Received Signal Strength

Scene analysis methods require significant initial effort to create the fingerprint DB

The RSS in indoor environments is influenced by:

- **multipath:** due to reflected signals, measured strength is higher than ideal
- **shadowing:** the signal strength in case of NLOS (non-line-of-sight) is not easily computable as the transmitted frequency is absorbed also by water (and hence by people)
- **moving objects:** cause sudden oscillations of the RSS thus requiring multiple measurements

In case of **non-temporary variations** of the environment (furniture reorganization or renewal) or modified position of the BS's we need to perform again the **offline training phase**

k-Nearest Neighbour (kNN)

Given:

- m the number of BS's
- n the number of fingerprints in the training set
- S_i the fingerprinting (array of cardinality m) corresponding to the point (x_i, y_i, z_i) of the training set
- s the measurement of the RSS as performed by the MS in the online phase

The kNN algorithm computes the distance s from every fingerprint in the training set; the chosen k points in the training set are those with the smallest d_i^2 values

MS coordinates are then estimated as the arithmetic mean of the coordinates of the aforementioned k locations or as the or weighted mean of the aforementioned distances

RFID (Radio Frequency Identification)

RFID systems are composed of:

- **RFID reader:** a device emitting a signal to query tags in proximity and receive the ID of each tag as a response
- **RFID tag:** a device that answers to the query of the reader by transmitting its own ID
 - Passive tag: low cost (0.3\$), shorter range, longer duration (no battery)
 - Active tag: higher cost (3\$), longer range, shorter duration (battery)
- The more expensive component of this system is definitely the reader (around 1000\$)

LANDMARK

LANDMARC is a positioning system that exploits active RFID.

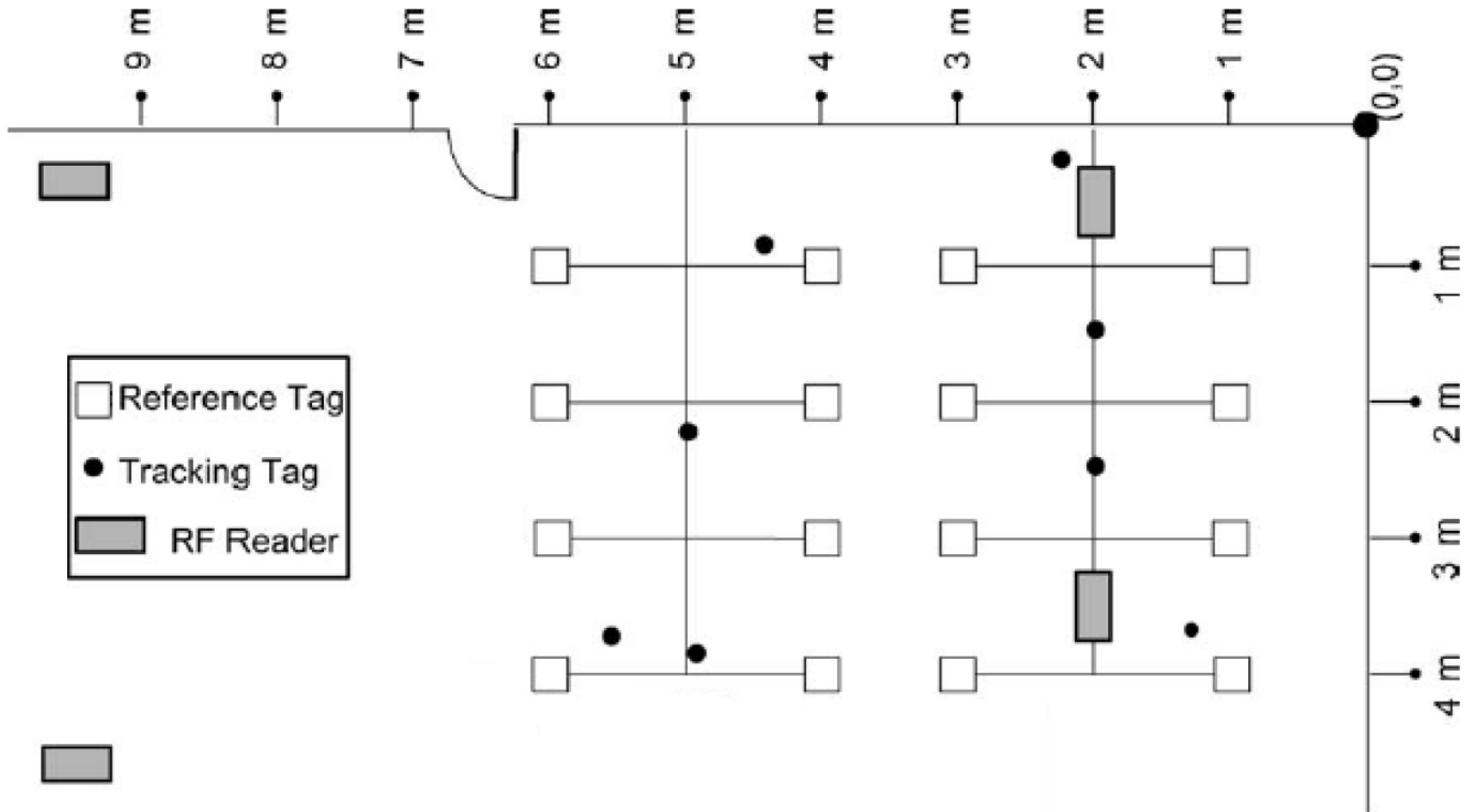
It uses a scene analysis method based on RSSI and composed of:

- RFID readers: used as BS with the capability to communicate performed measures to a localization server
- Reference tag: tag with known coordinates
- Tracking tag: tag to be localized (MS)

In this system a fingerprint is the array of RSSI of the signals emitted by RFID tags and received by the BS.

Thanks to the use of the reference tag the offline phase to train the system is not needed: the fingerprints related to the position of these tags are dynamically measurable making the system robust against scene changes

LANDMARK



LANDMARK

- For the localization the kNN algorithm is used exploiting the reference tag fingerprints as training set
- The system is significantly affected by the employed hardware as not all RFID readers provide a sufficiently fine granularity of the RSS
- The active RFID tags are powered by a battery and the system requires that the transmission power of all the tags (reference and tracking) be very similar
 - We hence need to use RFID tags of the same type and with the same level of battery to obtain comparable measurements that can be used

Passive RFID

The use of a RFID reader as MS is less expensive (than the previous solution) when:

- we have a wide space
- we need to localize few nodes

The use of passive RFID tags as reference tags reduces the cost
It is based on fingerprints and the training phase is expensive (2000 snapshots for 50 m²).

- A snapshot is obtained from different measurements of the same location (multiple reads are needed)

Not very robust against variations o the environment and adequate mostly for robotics/automated environments

Comparison of Legacy Solutions

System	Acc.(m)	Prec.(%)	Rob.	C. I.*	C. MS
802.11					
RTT SW 4w	3.5 - 4.2	50 - 90	3	280	600
RTT HW Newton	2.8	90	3	280	300
RTT HW Kalman	0.9 - 1.4	66 - 90	4	280	300
TDOA 11 BS	2.4	67	3	970	300
RSS k-NN	2.5 - 5.9	50 - 90	2	280	300
RFID					
LANDMARC	1 - 2	50 - 100	5	3600	10
Passive	0.55	90	1	240	800

Accuracy

Precision

Robustness

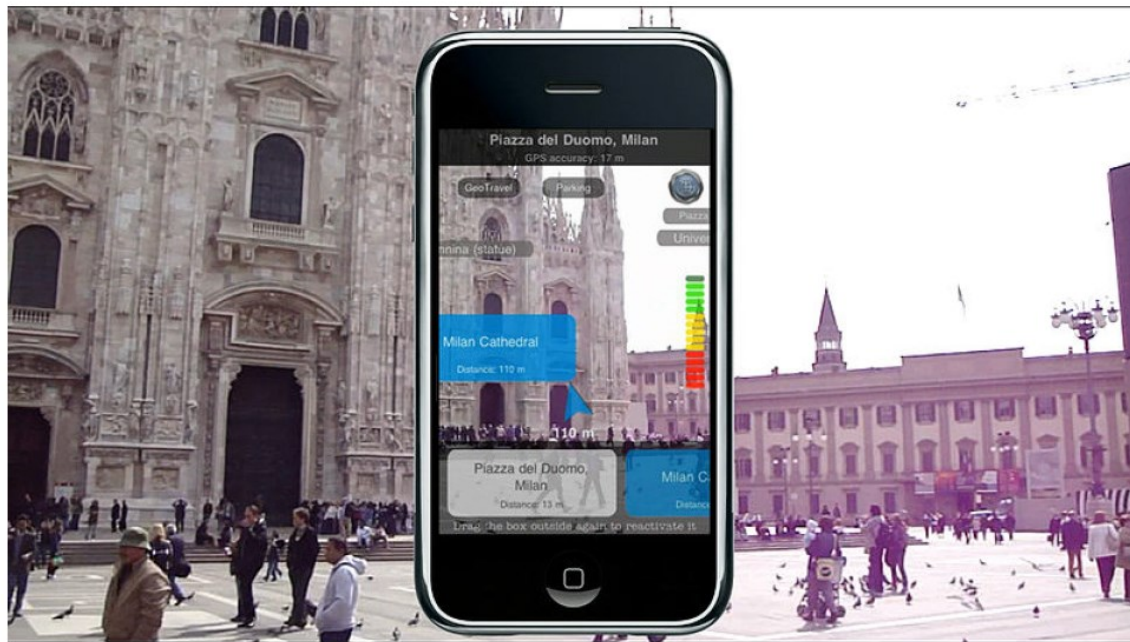
Cost-of-Infrastructure (* for a 400 m² environment: 20 x 20)

Cost-of-MS

How about Visual/AR Solutions?

Augmented Reality is based on the superimposition of informative levels (virtual, multimedia, geolocalized elements) to the real world.

Location-aware applications that exploits GPS, compass and accelerometer of mobile devices already exist.

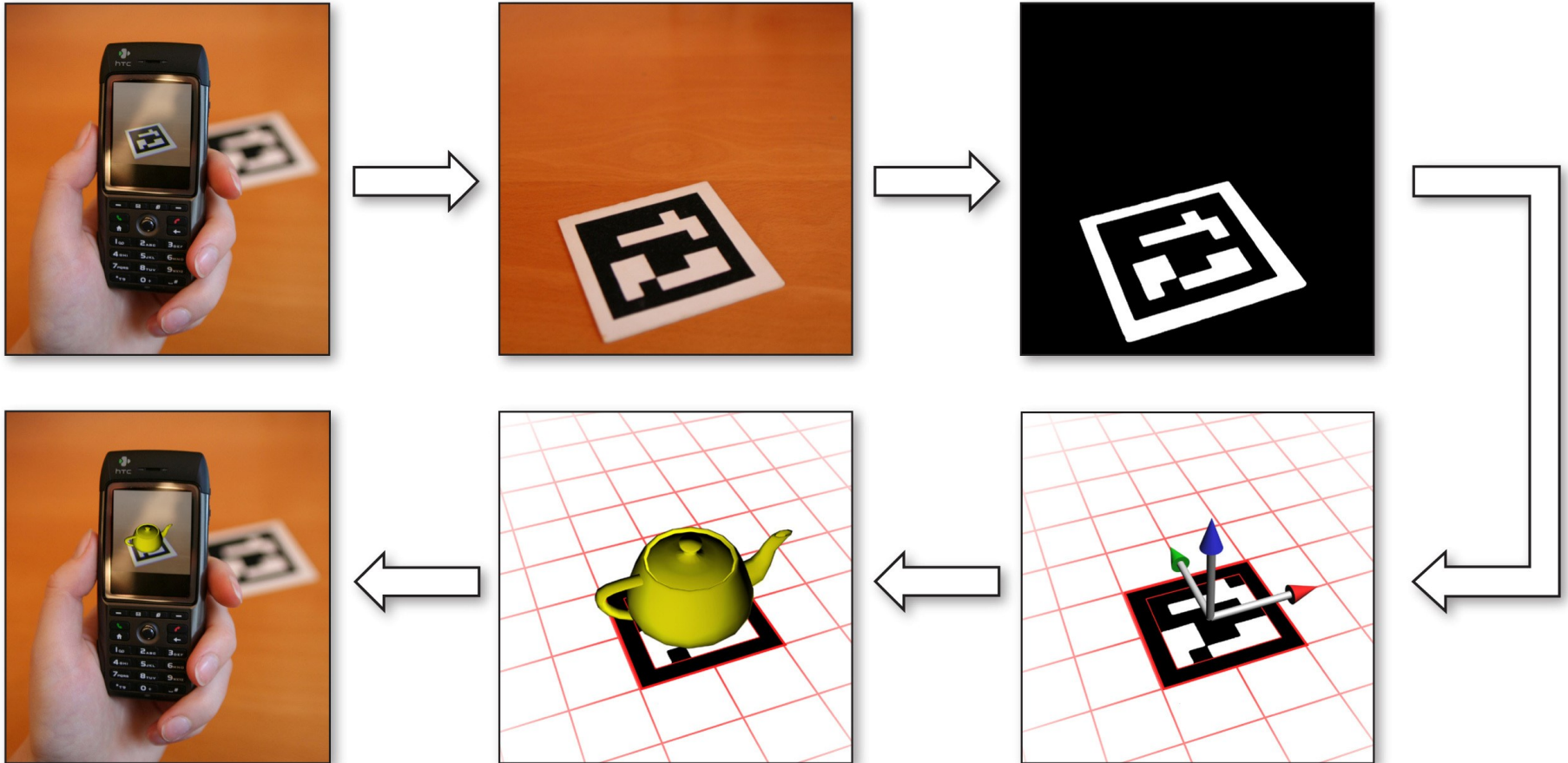


Visual Markers

- The position extraction through artificial markers is used mainly in AR
- To superimpose virtual contents to a real image we need to know with a good level of precision the actual position of the markers in the reference model of the camera.
- ARToolKit is a reference library in this context
 - Developed by Hirokazu Kato



ARToolkit Functioning



Coordinates Translation

The position of an object (and hence of a reference system) in a 3D cartesian space can be described through a matrix:

$$M = \begin{bmatrix} R_{11} & R_{12} & R_{13} & T_x \\ R_{21} & R_{22} & R_{23} & T_y \\ R_{31} & R_{32} & R_{33} & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where R is the rotation matrix that describes the e la matrice di rotazione che describe the orientation of the object and T is the translation array

To extract the new coordinates (x_m, y_m, z_m) of a point $P(x, y, z)$ in the reference system defined by M a matrix product is sufficient

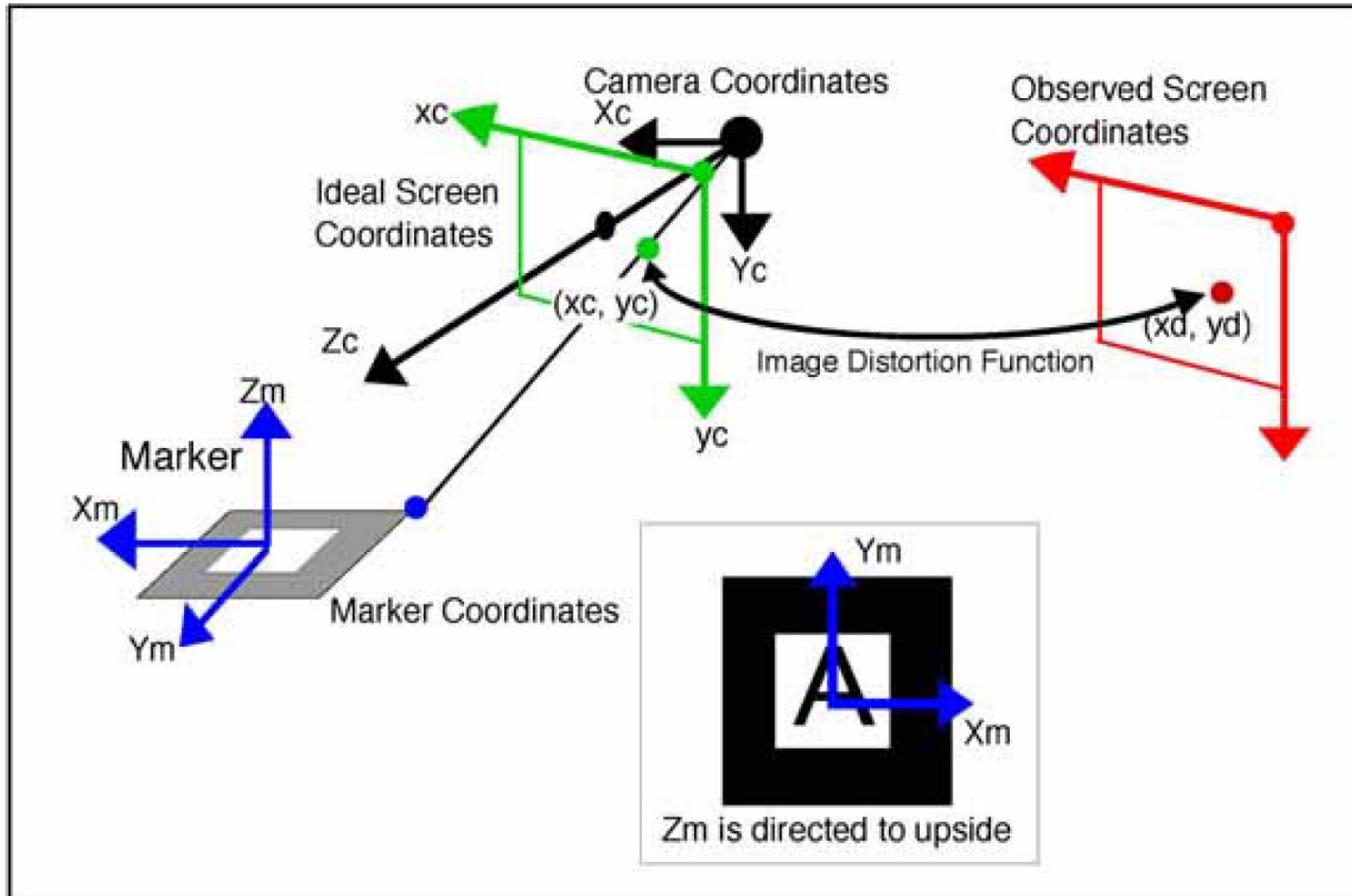
$$\begin{bmatrix} x_m \\ y_m \\ z_m \\ 1 \end{bmatrix} = M \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Coordinates Translation

- ARToolKit provide a the matrix P_m that represents the position of the marker in the camera reference system
- Matix P_c that describes the position of the camera in the marker reference system can be obtained as $P_c = P_m^{-1}$
- From P_c we can extract the translation array T_c
- Since the position of the marker in the global reference system is known, we also know G : the matix to translate the coordinates from the marker reference system to the global one

$$\begin{bmatrix} x_g \\ y_g \\ z_g \\ 1 \end{bmatrix} = G \cdot T_c$$

Reference Systems



Coverage of the System

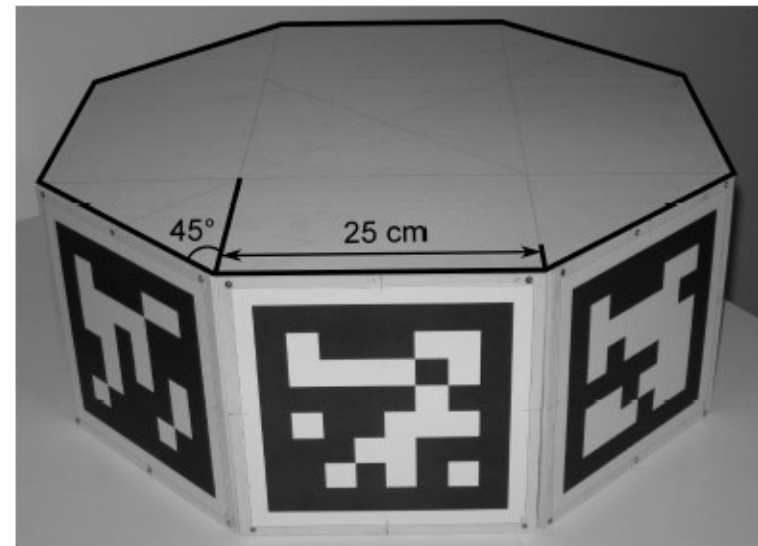
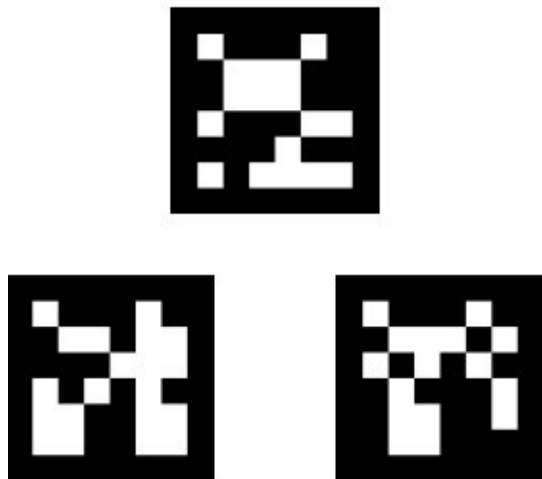
- With a regular webcam (video 640 x 480), a 8 x 8 cm marker is recognized at a distance less or equal to 2 m
- This is not sufficient for the considered problem of indoor localization
 - We would like to reach a distance of 10 m
 - A 50 x 50 cm marker would be needed
- The adopted solution was to improve the image resolution in order to obtain a good tradeoff between marker size and system coverage
- With 5MP images a marker of 20 x 20 cm can be recognized at a distance of 11 m

Error Sources

- Smartphones do not generally allow to access the raw version of the captured image, but only to compressed images (JPEG)
- JPEG compression is **lossy** (introduces noise in the input)
 - We can set the minimum level of compression to still have an acceptable level of precision
- The camera optics (lenses) distort the image, thus making necessary a calibration phase
- The **calibration** of a camera is a procedure that allows to extract in a semi-automatic way parameters that characterize its optics
 - Thanks to these parameters it is possible to compensate for the errors introduced by the optics
- The error on the position estimation is **proportional to the distance** due to the error in the orientation computation

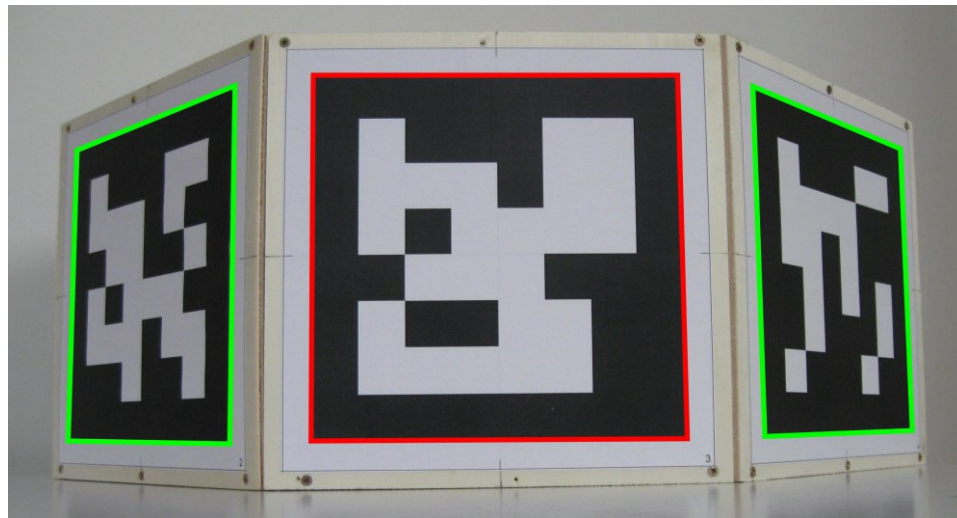
Multimarkers

- Experiments with single marker/tag results in too many errors
- Less errors if markers appear not frontally w.r.t. the camera
- Multimarker is a set of single markers arranged in a known way



Algorithm for Octagonal Multimarker

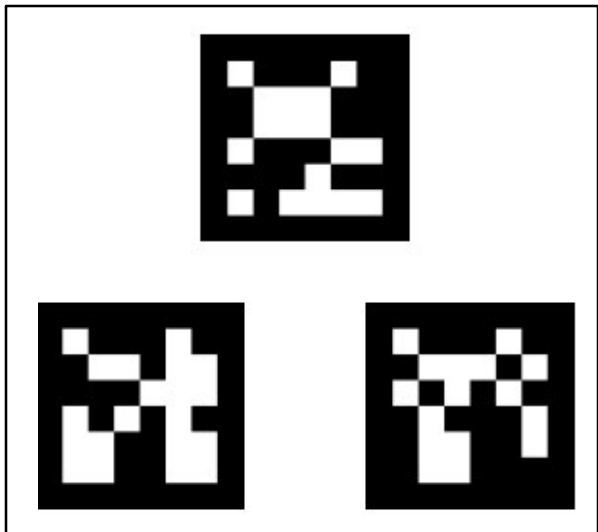
- The octagonal multimarker is made of markers on different planes and does not allow for the direct use of RPP algorithm
- Developed algorithm:
 1. Use RPP to estimate the camera position with respect to the single markers
 2. Assign a weight to each estimation, privileging non-frontal markers
 3. Compute the position of the camera as weighted average of the estimations



Experimental Setting

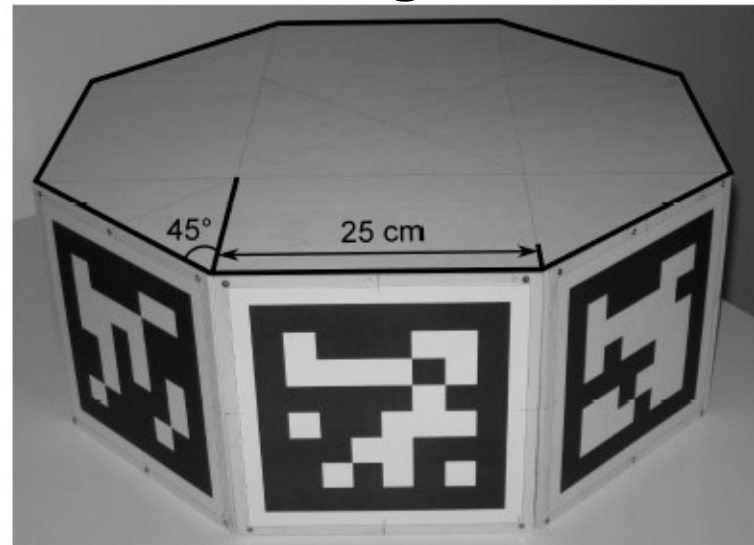
- Multimarker (single markers of 20 cm each)
- 18 pictures rapidly taken at 5 MP for each measurement location

Planar



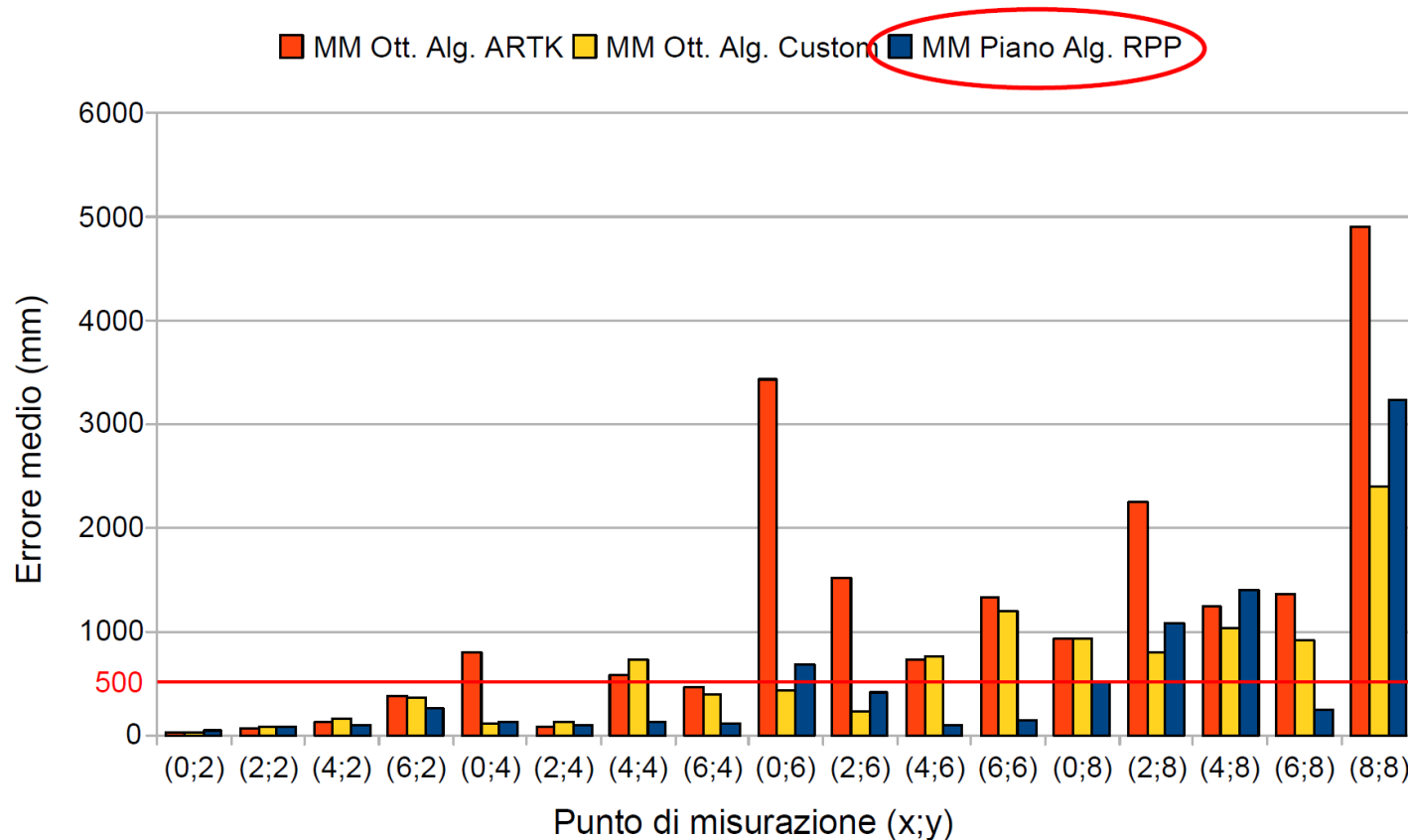
VS.

Octagonal



Results

- Red and yellow histograms refer to Octagonal multimarker
- Blue histograms refer to planar multimarker
 - Planar is simpler and not worse

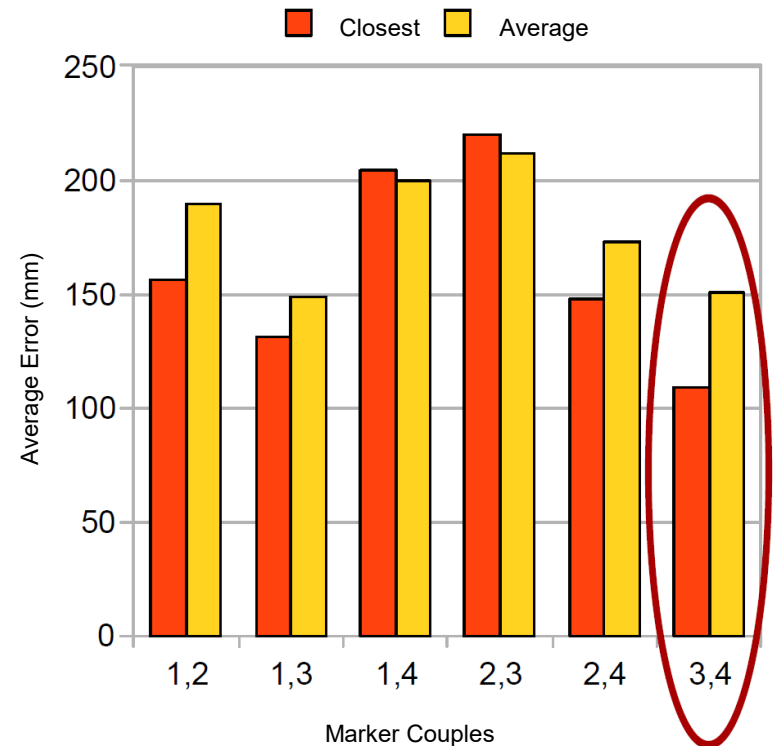


Prototype

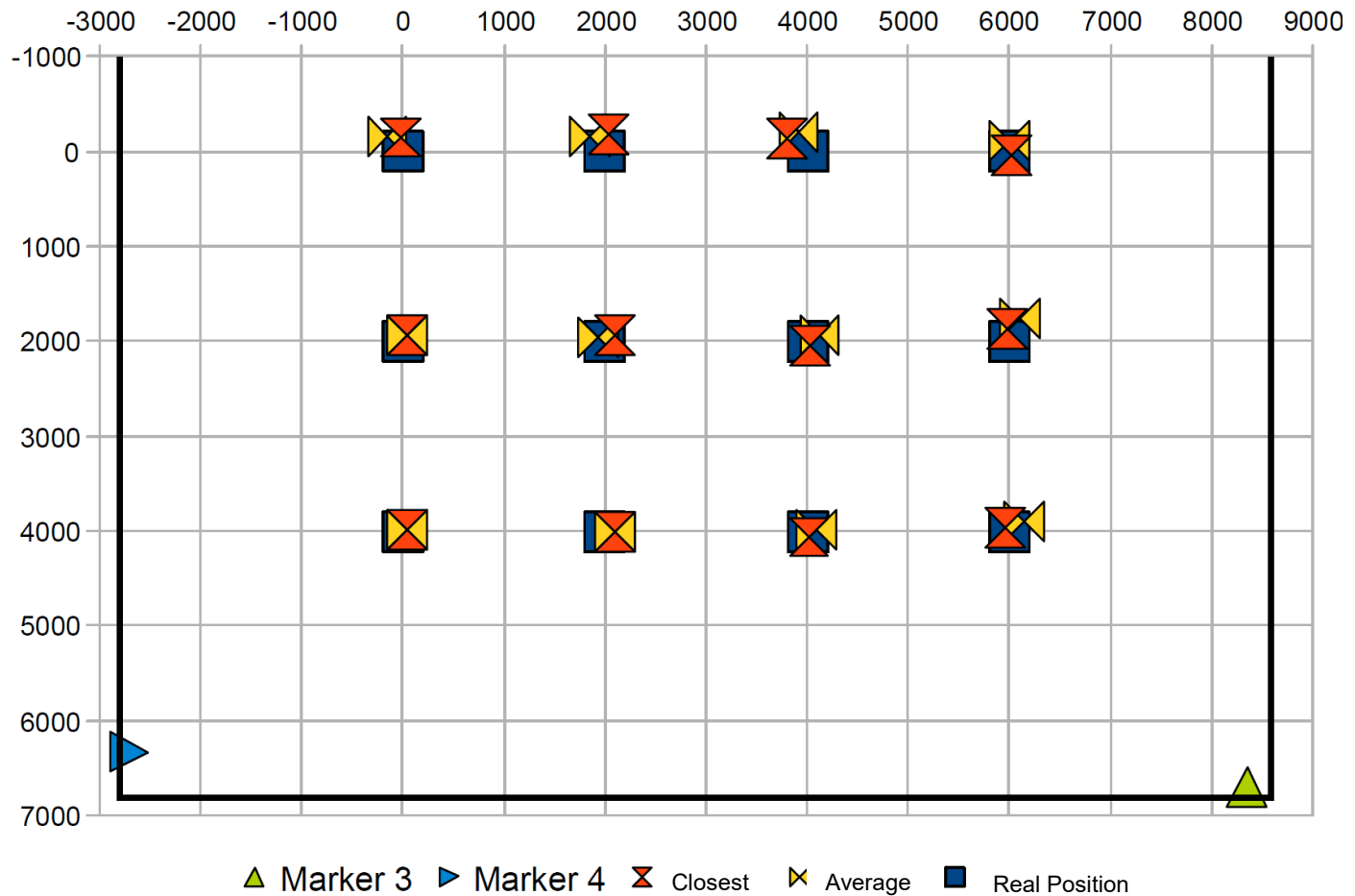
- Execution flow:
 1. Compute a set of estimations as a sequence of pictures
 2. Remove from the set the outliers
 3. Compute the estimated position of the camera as the average of the estimations left
- Experiment:
 - Real LuF2 room (9 x 11 m)
 - 2 or 4 planar multimarkers in proximity of the corners of the room
 - 18 rapidly shot images at 5 MP for each measurement

Results

- Two multimarkers are enough
 - no need for having one at each corner
- Compared strategies
 - Consider only the closest multimarker
 - Consider the average of the two estimations
- Average error
 - **Closest:** from 11 to 22 cm
 - **Average:** from 15 to 21 cm



Comparing Strategies for (3, 4)



Conclusion

- Advantages with respect to radiowaves systems
 - Higher precision
 - 22 cm vs 50 cm (for active RFID) vs 90 cm (for 802.11 trilateration)
 - Lower cost for the hardware
 - Lower installation and configuration time with respect to scene analysis methods
- Disadvantages
 - Line of sight should be free between the smartphone and at least one marker
 - Semi-automatic system

References

-  M. Ciurana, F. Barcelo-Arroyo, and S. Cugno,
“A novel TOA-based indoor tracking system over IEEE 802.11 networks” ,
in Mobile and Wireless Communications Summit, 2007. 16th IST, 2007, pp. 1 –5.
-  Lionel M. Ni, Yunhao Liu, Yiu Cho Lau, and Abhishek P. Patil,
“Landmarc: Indoor location sensing using active RFID” ,
Wireless Networks, vol. 10, pp. 701–710, 2004.
-  “ARToolKit online documentation” ,
www.hitl.washington.edu/artoolkit/documentation/.
-  C.B. Owen, Fan Xiao, and P. Middlin,
“What is the best fiducial?” ,
in Augmented Reality Toolkit, The First IEEE International Workshop, September 2002, pp. 98 – 105.